the Technical
Broadcast

Published by the Department of Information Services January/March

New Disaster Recovery Technology

Sterling Forest ... Now better than ever ♦

by Ken Boling

IBM's Sterling Forest Business Recovery Services Center has been the primary DIS S/390 recovery site since it opened in September 1993. However, as DIS needs grow and new technology emerges, DIS is fast outgrowing this site. DIS continues to investigate other possible sites to ensure that the site chosen is in the best interest of DIS and its customers. Preliminary planning indicates DIS will relocate to IBM's site at Gaithersburg, Maryland.

IBM continues to improve and enhance the facilities and systems of their Business Recovery Services Centers. Highlights of these improved enhancements at Sterling Forest include:

Infrastructure:

- Installed two 2,600 gallon potable water tanks. These tanks will supply water to the site should city water not be available due to a water main break. They also have redundant water systems that support chillers and condensers.

- Added air handling equipment and additional chilled water pumps to increase the cooling capacity within the

- Improved redundancy of electrical systems within the site with the addition of a larger bus for power distribution.

Network:

- Expanded network area by 6,000 square feet of raised floor allowing more room for customer equipment and future growth.

- Increased channel switch capability by 50 percent.

- Doubled channel extension equipment to enhance the capabilities of their remote customer sites.

Systems:

- Added a high-speed link to the Tandem™ systems allowing remote connectivity to host systems anywhere in the country.

Additional Business Recovery Services (BRS) Enhancements:

- Installed additional equipment and raised floor to support a variety of customer vaulting requirements.

Inside...

New Disaster Recovery Technology

Sterling Forest ... Now better than ever

(Continued from page 1)

- Expanded the end-user recovery area by 25 percent to enhance support for client/server customers.

Gaithersburg Broadens Support

The Disaster Recovery Center in Gaithersburg, Maryland went into full operation on July 3, 1994. By October 1995, if all goes as planned, Gaithersburg will become the DIS S/390 primary recovery site. Designed for customers seeking a "mega hot- site" environment, this center features multiple ES/9000 processors with 1,100 MIPS capacity and five terabytes of DASD.

BRS at Gaithersburg can support IBM's complete family of high end processors due to the installation of the 9672 Parallel Transaction Servers, models E01 through E04. Features of these servers include:

- One to four gigabytes of storage
- 24 to 96 ESCON channels
- Six processors per central electronic processing complex
- Four to 16 ISC links
- Sysplex timer

In addition, Solution Validation Services Center in Gaithersburg can provide machine time, expertise, and project support for one-time or periodic testing.

Examples of Uses:

- Host application stress testing
- Client/server application testing
- End-to-end application testing
- Performance benchmarking
- Batch testing/batch window sizing
- IPL tests
- Push/pull testing
- Application tuning
- Testing of technology exploitation:
    - RAMAC DASD
    - Parallel processing
    - Dynamic I/O
    - EMIF
    - ESCON
    - Remote-ops
    - Sysplex
    - Eight-way processors

Machine Configurations Available:

- 9672-E01
- 9672-E02
- 9672-E04
- 9672-R61
- 9021-982
- 9021-941
- 9121-742
- Client/server LAN lab (100+ 486 processor workstations)
- 3990-2,-3,-6
- 3390/3380
- 3490/3490E/3480
- ESCON
- 3745s including 900 frame
- 3172s including TCP/IP offload

Disaster recovery planning for Washington State Government's critical business service is becoming increasingly more important. As state employees, it is important that we understand the services and capabilities of business recovery services and support Washington State's Disaster Recovery program.

OTS1100 to be Installed for Unisys Transaction Environment ◆

by Steve Chiechi

The Department of Information Services (DIS) is testing level 6R4 of the Online Transaction Security (OTS1100) system from Formula Consultants Inc. DIS plans to have OTS1100 installed and available for all transactions in all application groups (“partitions”) on the Unisys 2200 development and production systems by mid-summer 1995.

Background

Since the installation of Unisys (then Sperry) mainframes at DIS in the mid-1970s, the transaction processing (TIP1100) environment has been secured through locally written and maintained programs. Customer security personnel allow and disallow execution of transactions based on terminal identifier (PID), operator number/password, time of day, timeout, and other parameters.

While this system works well, it is written in the meta-assembler language (MASM) of the 1100/2200 mainframes and is reaching the end of its useful, maintainable life. The number of different transactions that can be included in the system is currently limited to 1,500. Expanding this number and adding functions now required by the State Auditor’s Office represent a development and maintenance effort that would not be cost-effective given the availability of the vendor-maintained OTS1100 product.

OTS1100 Features

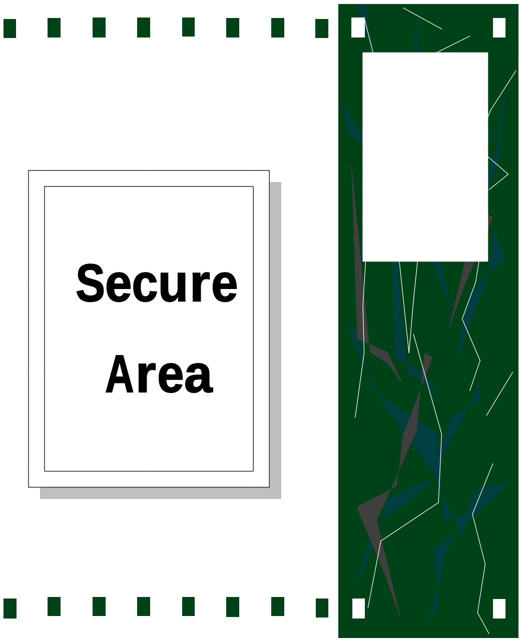
With the exception of time-of-day security, OTS1100 provides all the above functions. In addition to support for an adequate number of transactions, terminals and operators, new features include requiring 5-character-or-greater passwords and password aging.

Implementation

In order to fully utilize the features of OTS1100, transactions must be re-mapped using the OTS1100 versions of message handling routines.

The current transaction security system will remain in place to secure transactions until some future date.

Please direct any questions to Steve Chiechi at (360) 902-3044.
◆



Software AG’s PREDICT Version 3.2.3

◆ by Tom Thomas

The Department of Information Services (DIS) recently converted all customers’ PREDICT data dictionary systems to version 3.2.3. This is Software AG’s (SAG) newest release which provides additional functionality and enhancements and also fixes known problems from the prior release.

During the recent conversion, DIS discovered a situation that is related to the customer PREDICT administration practices. Some of our customers only maintain one PREDICT data dictionary system in their test or development environment. They create an ADABAS file or userview in test PREDICT and then generate a Data Definition Module (DDM). They then use the SYSDDM or SYSMAIN utility to copy the DDM to their production environment. The problem with this practice is that the production PREDICT does not recognize this DDM as one that it has generated and it will NOT migrate when we convert to a new release of PREDICT.

SAG recommends that you always copy the file or userview along with the DDM using the batch migration utility. The jobstream below is an example of migrating a file and DDM from a test environment to production. The CODE=Y parameter in STEP1 tells the migration utility to include any generated code such as DDMs during the UNLOAD. Following the recommended procedure will guarantee that all of your DDMs get migrated during future PREDICT upgrades. If you choose to continue your current PREDICT administration practice, keep a documented list of your DDMs so you can copy them

```
//J O B       C A R D
/*ROUTE  PRINT LOCAL
//*********************************************************
//STEP1    EXEC NAT,DBID='230'
//CMWKF01  DD  DSN=YOUR.FILE.NAME,DISP=(,CATLG),
//             DCB=(LRECL=4624,BLKSIZE=4628,RECFM=VB),
//             UNIT=SYSDA,SPACE=(CYL,(1,1))
LOGON SYSDICBE
MENU
UNLOAD,FILE TOMS-TEST-FILE,CODE=Y
FIN
//*********************************************************
//STEP2    EXEC NAT,DBID='226'
//CMWKF01  DD   DISP=SHR,DSN=YOUR.FILE.NAME
LOGON SYSDICBE
MENU
LOAD ALL,REPLACE=Y,CODE=Y
FIN
```

Software AG’s PREDICT Version 3.2.3

(Continued from page 4)

The old PREDICT version 3.1 reference and administration manuals are not compatible with the new release. You can order the PREDICT version 3.2 Reference and Administrator’s manuals directly from SAG. They also provide a PREDICT 3.2 Concepts and Facilities manual. It is intended to provide both prospective and current users with a general understanding of the wide scope of facilities and functions provided by PREDICT.

SAG only accepts orders via mail or fax from authorized users. To qualify as an authorized user, SAG requires that you provide them with a customer number which can be obtained from the DIS Data Base Support Group (360) 902-3135. Please provide the following information when ordering documentation:

- Customer number
- Customer name, address, and phone number
- Terms of payment (purchase order or credit card number)
- Specify air shipment (if needed)
- Specify title, manual number, price, and quantity of documentation

The following information provides the title, manual number, price, and the address to order documentation:

Title:	**Manual Number:**	**Price:**
PREDICT Administrator’s manual	PRD-320-040	50.00
PREDICT Reference manual	PRD-320-030	140.00

Mail order to:

Software AG
11190 Sunrise Valley Drive
Reston, VA 22091

Attn: Documentation Orders

FAX: (703) 391-6910

To assist you with your order, DIS Data Base Support Group can provide you with a Documentation Order Form.

If you have any questions about the new release of PREDICT, please call the DIS Data Base Support Group at (360) 902-3135. ◆

CICS - Why did the Program Fail?

◆by Carol Criscione

How many times do you receive a call for assistance with a program and the only symptom you are given is that the "program doesn't work?" Often there is some sort of output, error message, or abend code that assists in determining the cause of the problem. Sometimes there is little data so debugging the problem can be very time consuming.

I write my applications so that many of the errors and conditions are handled within the application itself. That allows me, and anyone else who may "inherit" my application, to diagnose a problem easily. I use the CICS-returned data contained in the Execute Interface Block (EIB).

Each time CICS performs an EXEC CICS command, a condition is raised. By using the RESP and RESP2 options in the command, I can determine my course of action. I normally code the error-handling routine immediately after my EXEC CICS command for "run-time" efficiency. I also define a variety of literals so I can be explicit about the error. A little extra time and care when writing the application can pay off.

On pages 125-128 of the CICS V3+ *CICS/ESA Application Programming Guide* (SC33-0675) there is an overview of handling exceptional conditions and reference to additional documentation.

The CICS V3+ *CICS/ESA Application Programming Reference* (SC33-0676) lists allowed CICS commands in alphabetical order. The "Command Syntax" section contains the command syntax, conditions that are possible upon executing the command, a brief description of the command, and a brief description of each condition associated with the command including any applicable RESP2 codes. Pages 316-317 summarize the EIBRESP numbers and conditions.

The following is an *abbreviated* sample of one way to code handling exceptional conditions after executing an 'XCTL' command in COBOL/370:

```
WORKING-STORAGE SECTION.
77  RESPONSE-1   PIC S9(8)  VALUE ZEROS
    COMP.
77  RESPONSE-2   PIC S9(8)  VALUE ZEROS
    COMP.

*ERROR MESSAGES
01 WS-AREA.
   05  WS-ERRMSG-INVALID
       PIC X(32)  VALUE 'INVALID REQUEST
       HAS BEEN ISSUED.'.
```

(Continued on page 7)

CICS - Why did the Program Fail?

(Continued from page 6)

```
     05  WS-HELP
         PIC X(26)  VALUE 'CALL FOR HELP AT
         XXX-YYYY.'.
     05  WS-ERRMSG-PGMID1
         PIC X(35)  VALUE 'THE PROGRAM IS
         NOT DEFINED TO CICS.'.
     05  WS-ERRMSG-PGMID2
         PIC X(24)  VALUE 'THE PROGRAM IS
         DISABLED.'.
     05  WS-ERRMSG-PGMID3
         PIC X(29)  VALUE 'THE PROGRAM
         CANNOT BE LOADED.'.

PROCEDURE DIVISION.

(BEGINNING CODE TO XCTL COMMAND)

*XCTL TO PROGRAM NUMBER 5
   400-XCTL-TO-C999PROG.
       EXEC CICS XCTL
          PROGRAM('C999PROG')
          COMMAREA(MYAREA)
          RESP(RESPONSE-1)
          RESP2(RESPONSE-2)
       END-EXEC.

*IF NORMAL RESPONSE SKIP PAST ERROR
HANDLING
   IF RESPONSE-1 = DFHRESP(NORMAL)
       GO TO 500-CONTINUE-ON.

*PREPARE ERROR MESSAGE OUTPUT
   MOVE SPACES TO DISPLAY1 DISPLAY2.
   MOVE WS-HELP  TO DISPLAY2.

*INVALID REQUEST?
   IF RESPONSE-1  = DFHRESP(INVREQ)
   MOVE WS-ERRMSG-INVALID TO DISPLAY1
   GO TO 450-RETURN-ERR-MAP.

*PROGRAM  ERROR?
        *PROGRAM NOT DEFINED?
   IF RESPONSE-1  = DFHRESP(PGMIDERR)
       IF RESPONSE-2 =  1
           MOVE WS-ERRMSG-PGMID1 TO
           DISPLAY1
           GO TO 450-RETURN-ERR-MAP
       ELSE
          NEXT SENTENCE
   ELSE
          NEXT SENTENCE.

*PROGRAM  ERROR?
        *PROGRAM DISABLED?
   IF RESPONSE-1  = DFHRESP(PGMIDERR)
       IF RESPONSE-2 =  2
           MOVE WS-ERRMSG-PGMID2 TO
           DISPLAY1
           GO TO 450-RETURN-ERR-MAP
       ELSE
          NEXT SENTENCE
    ELSE
          NEXT SENTENCE.

*PROGRAM  ERROR?
        *PROGRAM CANNOT LOAD? (DEFAULTS
         TO GENERAL ERROR)
   IF RESPONSE-1  = DFHRESP(PGMIDERR)
       IF RESPONSE-2  = 3
           MOVE WS-ERRMSG-PGMID3 TO
           DISPLAY1
           GO TO 450-RETURN-ERR-MAP
       ELSE
           MOVE 'THIS IS A GENERAL PROGRAM
           ERROR' TO DISPLAY1
           GO TO 450-RETURN-ERR-MAP
    ELSE
          NEXT SENTENCE.

*LENGTH ERROR?
*DEFAULTS TO MISC ERROR
   IF RESPONSE-1  = DFHRESP(LENGERR ...
   ELSE
        PERFORM...

 *RETURN TO DISPLAY AND CURRENT
TRANID
   450-RETURN-ERR-MAP.
       EXEC CICS SEND  MAP('M999PGM')
           ALARM
```

(Continued on page 8)

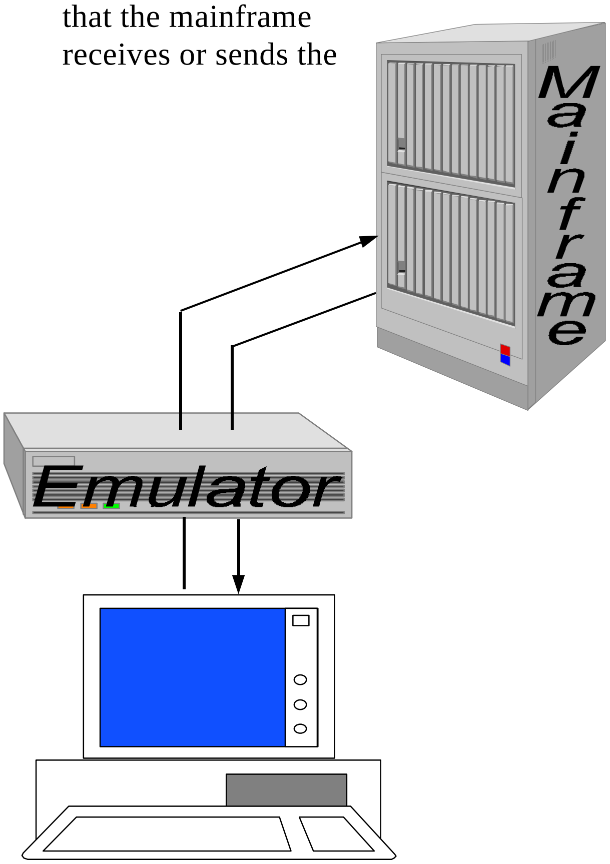
PC/Mainframe File Transfer

◆ **by Judy Holm**

Transferring files between the mainframe and PC has become a common way to get information where it needs to be. Usually, file transfer is reliable and easy to use. But sometimes problems arise that are hard to find. With all the different layers of communication and protocol, it is often difficult to determine where the problem lies.

It is difficult to learn every type of protocol and communication that lies between your PC and the S/390 mainframe. The following file transfer elements we can help you with are:

1. Make sure the mainframe is working properly. We will do a file transfer from our PC to the mainframe to ensure that the mainframe receives or sends the



2. If you are using EXTRA! from Attachmate, validate that the parameters you are entering are the correct parameters for your application.

3. If you are not using the Attachmate emulator, and the mainframe connection is working properly, contact your LAN coordinator or the vendor of the emulator package you are using for assistance. ◆

CICS - Why did the Program Fail?

(Continued from page 7)

```
      ERASE
      FREEKB
    END-EXEC.

    EXEC CICS RETURN
      TRANSID(EIBTRNID)
      COMMAREA(MYAREA)
    END-EXEC.
  450-RETURN-ERR-MAP-EXIT.  EXIT.

  500-CONTINUE-ON.
(CODE  CONTINUES)
```

Get the idea? I find using the EIB data for exceptional condition processing makes my “debugging experience” easier. I hope you’ll find it helpful, too! ◆

CICS COBOL/370 Compiler Options Part II ♦ by Gayle Huck

Part I of this article, published in the October/December 1994 issue of the *Technical Broadcast*, pointed out that certain compile options are more effective in the test environment and others are more effective in the production environment. It also explained that if you use the CICS PROC CICSVCCL, you may use the symbolic parameter CPARM to override compile options. You must be aware that overriding any compile option required by XPEDITER/CICS will result in a return code of eight in CICSVCCL's XPEDITER/CICS postprocessor step.

Following is a list of common compile options with a brief description of their use:

APOST
This is both a translator and compiler option and requires all literals be enclosed within single quotes. It is an installation default due to customer request. This compile option may be overridden if an agency decides to adhere to the ANSI85 standard. QUOTE, the option that adheres to the ANSI85 standard, requires all literals be enclosed within double quotes. The option must be the same on the CICS translator step.

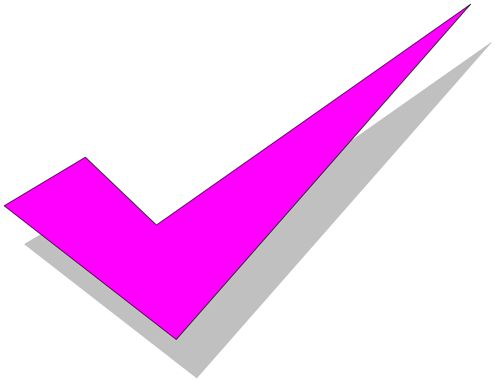
LIST
This option, used in the CICSVCCL PROC, generates assembler code for the compiled program, global tables, literal pools, and information about working-storage. **LIST must be used with XPEDITER/CICS if OPTIMIZE** is used. (See NOOPTIMIZE and OPTIMIZE sections for more information.) LIST may be overridden with NOLIST which is the installation default.

MAP
This option, which **must be used in the test environment with XPEDITER/CICS**, produces a listing of the items defined in the Data Division and includes: Data Division map, global tables, literal pools, nested program structure map, program attributes, and size of the program's working-storage. MAP may be overridden with NOMAP.

NOOPTIMIZE
This option does not generate optimized code and should **always be used in the test environment** for debugging purposes.

XPEDITER/CICS recommends NOOPTIMIZE be used in order for all the debugging functions to work properly. These functions include execution trace, code stepping, and resuming execution at another statement. If XPEDITER/CICS is being used on a program that is not optimized, LIST or OFFSET, the installation default, may be used.

(Continued on page 10)



CICS COBOL/370 Compiler Options Part II

(Continued from page 9)

NOSEQUENCE
This option, an installation default, suppresses checking numbers coded in columns 1 through 6 of your source and issuing diagnostic messages if the source is not numbered in ascending sequence. Although NOSEQUENCE is more efficient, it can be overridden by SEQUENCE.

OFFSET
This option, an installation default not used in CICSVCCL PROC, generates a condensed listing and shows the beginning offset location of each procedural statement. It may be used with XPEDITER/CICS only if
NOOPTIMIZE is used (see NOOPTIMIZE and OPTIMIZE for further information). OFFSET may be overridden by NOOFFSET.

OPTIMIZE
This option generates optimized code. This option **must always be used for production** programs because it improves the run-time performance of COBOL/370 programs.

If XPEDITER/CICS is being used on a program compiled with OPTIMIZE, LIST must be used rather than OFFSET. LIST and OFFSET are mutually exclusive.

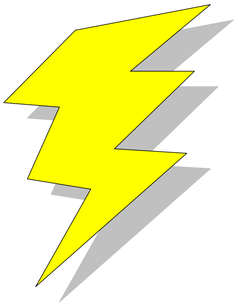
CICSVCCL uses LIST with the installation default OPTIMIZE in order to accommodate both the production and test environments. You will receive a return code of 4 in your compile due to the following warning

GYOS4020-W THE "OFFSET" OPTION WAS DISCARDED DUE TO OPTION CONFLICT RESOLUTION. THE "LIST" OPTION FROM COMPILER INVOCATION PARAMETER TOOK PRECEDENCE.

NOTE: Maintenance has been applied to COBOL/370 in which OPTIMIZE can either be OPTIMIZE(FULL) or OPTIMIZE(STD). CICSVCCL will use the installation default of OPTIMIZE(FULL) which improves the run-time performance. XPEDITER/CICS supports OPTIMIZE(STD) with the LIST option but it does not support OPTIMIZE(FULL) at this time. The vendor is working to provide this capability. NOOPTIMIZE is still the recommended option to use with XPEDITER/CICS. If you use OPTIMIZE with no suboption, you will receive OPTIMZE(STD). See DIS *Technical Bulletins* 1.2.10.22 and 1.2.3.10 for further information.

SOURCE
This option, an installation default, lists the source code in the compiled output and **must be used in test with XPEDITER/CICS**. It can be overridden by NOSOURCE.

(Continued on page 11)



CICS COBOL/370 Compiler Options Part II

(Continued from page 10)

SSRANGE
This option allows debug code for subscript and index range checking to be included in the object module. **It must only be used in test because it degrades performance and increases CPU usage during program execution**. SSRANGE also requires an installation run-time option be set to CHECK(ON) in order for it to take effect. CICS run-time option is set to CHECK(ON).

At run-time, checks are not made on each individual subscript or index but on the address to ensure it does not refer to an item outside of the table. The program will terminate with the following error message:

DFHAC2206 12:21:51 NCICCC Transaction TTTT has failed with abend 3000. Resource backout was successful.

To determine if this was an out-of-range condition, you must browse a transient data queue (sequential file), “CICS.LE370.CIC**.CEEMSG”, where ** will be the CICS/ESA region you are testing in (X1-X6). You must browse this queue for any abend that is 3000 and higher. To do so:

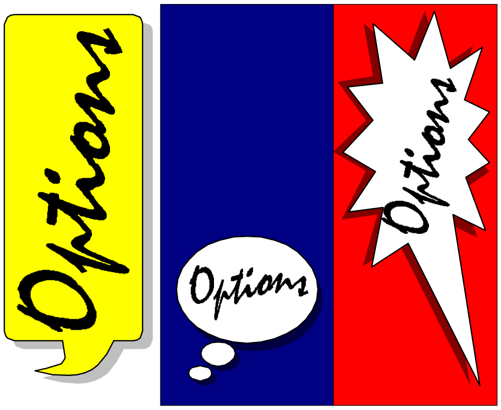
1. Close and open the transient date queue by the following command:

   ```
   CEMT S TD(CESE) CLO
   CEMT S TD(CESE) OPE
   ```

2. From Spiffy’s main menu, enter 1 on the option line to get to the browse panel. Browse CICS.LE370.CIC**.CEEMSG (** = X1 - X6). Within the dataset text, you should see an error message such as “IGZ0006S. The reference to table WS-TABLE by verb number 01 on line 000700 addressed an area outside the region of the table.”

WORD(CICS)
This option applies checks at compile time and restricts COBOL reserved words, such as Read, Write, and File, which are not supported by CICS. The Input-Output Section and File-Control in the ENVIRONMENT DIVISION are also disallowed. NOWORD suppresses this verification. CICSVCCL overrides the installation default, NOWORD, with this option.

XREF(FULL)
This option generates a cross-reference listing of all names and **must be used in test with XPEDITER/CICS**. NOXREF
su pp
re ss
es th
e lis
ti ng
. CI
C S
V C
C L
o ve



The WHO Report: Who did what with ADABAS ♦ by Keith Newsom

The Department of Information Services (nee WDPSC) installed ADABAS in October 1975. (No, not the one we’re running today. It was ADABAS 3, and NATURAL wasn’t even born yet.)

It wasn’t long before someone asked, “What happened with ADABAS yesterday?” Rather than give the old software shrug, I developed a report based on System Measurement Facility (SMF) step-end records that shows who did what with ADABAS. Just like ADABAS, the report has grown and changed over the years. It is very useful for answering questions like:

“How many job steps used ADABAS?” (Would you believe over 10,000 a day?)

“What system(s) did they run on?”

“Did any of them abend?”

“Was there a pattern of several abending at the same time?”

“Were there a lot of jobs running simultaneously against data base 193?”

“Did agency 999 run any ADABAS jobs on the Q1 system?”

The WHO report is run daily at approximately 2 a.m., after the SMF data is processed. DIS keeps the last 15 reports in a Generation Data Group named ‘DBUD155.WHO.’ Please feel free to browse it.

```
                     JOB STEPS USING DBID “193”
     FEBRUARY  20, 1995 (95051) 23:16 - FEBRUARY  22, 1995  (95053) 00:18
```

The date-time range is derived from the report and shows the earliest step start and the latest step end. You may have to look at two or more WHO reports to get all the information about a time period of interest. Remember that the data is derived from step-end records. If a step lasts 70 hours, it won’t be recorded until it ends.

The report is broken down by data base ID. Most people are interested in only one data base. An easy way to find it is to use a TSO command such as F ' “193 '

(Continued on page 13)

The WHO Report: Who did what with ADABAS

(Continued from page 12)

Next we have the meat of the report, one line per job step. (A TSO session is a one-step job.) Some of the columns describe a job as a whole:

```
JOBNAME  READER TIME    JOB/  C   . . .   S
                        TSU   L   . . .   ID
                        #     S
DH3AR303 95052 06:47:08 02259 A   . . .   Q2

MIXBA105 95052 07:00:18 02384 R   . . .   Q1
```

This really nails down what job it was -- the job name, the date and time it was read into the system, what JES number it was assigned, what class it was, and what system it ran on.

Other columns describe the steps that used ADABAS:

```
STP STEPNAME     PROGRAM         PROGRAM     PROGRAM  COND
 #               START           STOP        NAME     CODE

001 NATURAL  052 06:47:13.76 052 06:47:14.76 BNAT     S000

001 STEP01   052 07:00:23.16 052 07:00:24.30 DBSTATUS S000
002 STEP01A      07:00:26.12     07:00:31.80 BNATOPT  S000
005 STEP03       07:00:41.67     07:00:47.07 BNATOPT  S000
```

This gives the step number within the job, the step name, when the program started, when it stopped, the program name, and what condition code it set. Program names like BNAT and BNATOPT are really aliases, but they tell what cataloged procedure the step was using. The report is sorted by program start time within data base ID.

Note that the data base ID is that of the first data base accessed by the program. This results from some locally-written code linked to ADALNK, the batch/TSO-ADABAS interface module. This first-time-only code dynamically allocates a DDNAME of DBIDnnn, where nnn is the first data base you called. This DDNAME then shows up in the SMF record written at step-end time. Without it, we couldn’t isolate the SMF records that are ADABAS related.

Now you can do ADABAS post-mortems (a timely subject these days).
Happy sleuthing. ◆

the Technical Broadcast

Department of Information Services
1310 Jefferson ST SE
Mail Stop: 42452
Olympia, WA 98504-2452

Technical Broadcast Staff - The *Technical Broadcast* is published quarterly by the Department of Information Services (DIS). The purpose is to provide a forum for customer information-sharing of upcoming system enhancements and optimization tips. We invite your comments and suggestions.

If your name or address is incorrect, or if someone you know wishes to be added to or deleted from the distribution list, please fill out the form below, cut, and send via campus mail at the above address. Thank you.

Department of Information Services (DIS) is an equal opportunity employer and does not discriminate on the basis of race, religion, color, sex, age, nationality, or disability.

Name: ____________________________________

Agency: ____________________________________

Address:____________________________________

City:________ State:________ Zip Code:__________

Mail Stop:________ Phone No.:__________

Washington State Department of
Information Services